**Final Project Report for the Irrigation Recommendation System**

Raulie Raulerson

Version 1.2

December 6, 2017

Table of Contents

[1. Acknowledgements 3](#_Toc500314725)

[2. Project Summary 3](#_Toc500314726)

[3. Project Outcomes 4](#_Toc500314727)

[3.1 Work Products Produced and Project Outputs 4](#_Toc500314728)

[3.2 Completion of Work Products and Achievement of Project Outcomes 6](#_Toc500314729)

[3.3 Lessons Learned 6](#_Toc500314730)

[4. Conclusions 7](#_Toc500314731)

[5. Recommendations 7](#_Toc500314732)

[6. References and Acronyms 8](#_Toc500314733)

[6.1 References 8](#_Toc500314734)

[6.2 Acronyms 9](#_Toc500314735)

# Acknowledgements

I would personally like to acknowledge and thank Dr. Donna Lohr, Katie Hallas, Charles Barrett, and the staff at Skymind for their help in making this project possible.

Dr. Lohr has provided invaluable direction to me during the project and served as my mentor. Katie Hallas with the Florida Department of Agriculture and Consumer Services (FDACS) served as the client for the project. She provided informative and constructive feedback to the documents that were produced for the project. In addition, she was instrumental in helping me gather requirements for the project. Mr. Barrett, with the University of Florida’s Institute of Food and Agricultural Sciences was kind enough to share soil moisture sensor (SMS) and weather data with me. I also want to thank the contributors and developers at Skymind. The staff at Skymind created the DeepLearning4J application programming interface (API). They also shared, via Github, a number of extremely helpful examples. Without the guidance from these examples and the help from community members on Gitter, this project would not have succeeeded. Without the contributions from all the individuals listed above, this project would not have been possible.

# Project Summary

Having an adequate water supply is an important issue that all Floridians will have to address in the near future. One new technology that is an innovative tool for conserving water is the SMS. This technology has the potential to significantly reduce a user’s water usage, when irrigating, by allowing accurate measurement of the moisture content of the soil, how deep the water content has permeated, the soil’s temperature, and even the soil’s electrical conductivity (EC). One of the biggest challenges that threatens the use of this tool is the difficulty of interpreting the data produced by these sensors. This affects both agricultural producers and well informed urbanites.

This project’s focus will be on the interpretation and analysis of the data provided by the soil moisture sensor(s) and other site-specific monitoring devices (i.e., weather stations). The project will utilize a recurrent neural network to analyze input data from these sensors, and other devices, to provide a recommendation to the user about how best to operate their irrigation system. Ultimately, this project will produce a product that has a graphical user interface (GUI) that a user can easily navigate. The GUI will allow the user to input SMS and weather data and generate an irrigation recommendation report. The hope is that this software product will lessen the burden of adopting and utilizing SMSs.

# Project Outcomes

## Work Products Produced and Project Outputs

Thus far this project has yielded the development of the following work products/project outputs, which have been sent to the client and the project’s mentor for review:

* Project Definition Background – Semester 1
* Project Definition Section 1 – Semester 1
* Project Definition Section 2 – Semester 1
* Project Definition Section 3 – Semester 1
* Project Definition Section WBS – Semester 1
* Project Definition Background – Semester 1
* Project Definition Section 1 – Semester 2
* Project Definition Section 2 – Semester 2
* Project Definition Section 3 – Semester 2
* Project Definition Section WBS – Semester 2
* Project Status Update 1 – Semester 1
* Project Status Update 2 – Semester 1
* Project Status Update 3 – Semester 1
* Project Status Update 4 – Semester 1
* Project Status Update 5 – Semester 1
* Project Status Update 6 – Semester 1
* Project Status Update 1 – Semester 2
* Project Status Update 2 – Semester 2
* Project Status Update 3 – Semester 2
* Project Status Update 4 – Semester 2
* Project Status Update 5 – Semester 2
* Project Status Update 6 – Semester 2
* Project Status Update 7 – Semester 2
* Project Status Update 8 – Semester 2
* Progress Report 1 – Semester 1
* Progress Report 2 – Semester 1
* Progress Report 1 – Semester 2
* Progress Report 2 – Semester 2
* Risk Matrix
* Risk Management Document
* User Interface (UI) Design
* SBDTS
* SPMP
* SDD
* SRS
* Project Code
* Software Testing Document (STD)
* User’s Manual
* Client Acceptance Email (for the Work Products listed here)

The major outcomes achieved during this the project were a fully implemented (in the development environment), and tested, Irrigation Recommendation System with an easy-to-use Graphical User Interface (GUI), project code (and JavaDocs), software testing document (STD), software project management plan (SPMP), risk management plan, risk matrix, software design description (SDD), software requirements specification (SRS), and software budget and detailed timeline spreadsheet (SBDTS).

## Completion of Work Products and Achievement of Project Outcomes

The project outcomes and work products generated during the second phase of this project are listed above. These project outcomes and work products were achieved/completed by creating a detailed timeline (SBDTS) that noted which tasks needed to be completed during each week throughout the project. This timeline helped keep the project manager on task and aided in recognizing when he was falling behind in the project. Ultimately, it was recognized that the project was behind schedule later in the semester. The mentor also mentioned this. The project manager got back on schedule by working diligently over the last month of phase two of the project.

In addition, frequent communication with the client was instrumental to completing the work products (listed above) on time and to their satisfaction.

## Lessons Learned

One of the main lessons that was learned during this phase of the project was to not deviate from the detailed timeline that was prepared. Deviation from the timeline, for even one week, drastically knocked the project off course. This required the project manager to work many additional hours during the final weeks of the project.

Another lesson that was learned was that the detailed design outlined in the SDD needed to be adjusted during the construction phase of the project. This was due to methods needing to be overwritten and an attempt to have low coupling between classes. These issues weren’t realized in the initial design for the system.

In addition to these lessons, deployment issues weren’t considered when assessing risks and constructing the detailed timeline and budget. Unfortunately, these issues proved to be difficult to overcome due to a dependency issue with DeepLearning4J. The product works well in the development environment and it archives, the system just doesn’t have complete functionality. In the future, a greater amount of time and research needs to be invested in the design and the deployment phases of the project. This will ensure construction goes smoothly and extra effort isn’t needed to fix design problems and that the system can be successfully deployed on the client’s system.

# Conclusions

Both phases of this project were quite challenging. This was mainly due to the deviation from the detailed timeline. In addition to the timeline issue, there were some obstacles that were encountered during the construction phase of the project that weren’t recognized in the design. These obstacles included having to revise parts of the design to make the system work and to ensure the classes were loosely coupled. Though the final phase of the project proved challenging, it generated the system that was intended; however, deployment of the product ran into some problems and was not entirely successful. The system includes a simple GUI and a user’s manual for making use of the system much easier.

# Recommendations

Based on the work products and conclusions of phase two of the project, the system was constructed as designed in the SDD and SRS. If further time were to be devoted to this project, it would be well spent on fine tuning the recurrent neural network (RNN) that is implemented in the code. By fine tuning the RNN’s hyperparameters, it is possible to get a more accurate irrigation recommendation from the model. This tuning could be accomplished by utilizing several tools that DeepLearning4J has made available through their API. In addition to fine tuning the RNN, it would be ideal to successfully deploy the IRS product on the client’s system. The archived JAR file that was created has dependency issues that must be addressed in the next phase of the project. The client has accepted the constructed system, so no further recommendations are warranted.

# References and Acronyms

## References

**Project Definition Report**

|  |  |
| --- | --- |
| Version | 1.2 |
| Date | 10/5/2017 |
| Author | Raulie Raulerson |

**Software Requirements Specification (SRS)**

|  |  |
| --- | --- |
| Version | 1.7 |
| Date | 12/4/2017 |
| Author | Raulie Raulerson |

**Software Design Description (SDD)**

|  |  |
| --- | --- |
| Version | 1.3 |
| Date | 12/5/2017 |
| Author | Raulie Raulerson |

**Software Project Management Plan (SPMP)**

|  |  |
| --- | --- |
| Version | 1.5 |
| Date | 12/4/2017 |
| Author | Raulie Raulerson |

**Risk Management Plan**

|  |  |
| --- | --- |
| Version | 1.5 |
| Date | 12/4/2017 |
| Author | Raulie Raulerson |

## Acronyms

API – Application Programming Interface

FDACS – Florida Department of Agriculture and Consumer Services

GUI – Graphical User Interface

IRS – Irrigation Recommendation System

JAR – Java Archive

RNN – Recurrent Neural Network

SBDTS - Software Budget and Detailed Timeline Spreadsheet

SDD – Software Design Description

SDLC – Software Development Life Cycle

SPMP – Software Project Management Plan

SMS – Soil Moisture Sensor

SRS – Software Requirements Specification

STD – Software Testing Document

UI – User Interface